



UNIVERSIDAD MILITAR  
NUEVA GRANADA

# **APLICACIÓN VISUAL MODELUILDER PARA DETERMINAR EL ÍNDICE VULNERABILIDAD A EVENTOS TORRENCIALES**

**RUTH JIMENA VANEGAS CASTELLANOS**

**UNIVERSIDAD MILITAR NUEVA GRANADA  
POSGRADOS FACULTAD DE INGENIERÍA  
ESPECIALIZACIÓN EN GEOMÁTICA  
BOGOTÁ D.C, COLOMBIA**



UNIVERSIDAD MILITAR  
NUEVA GRANADA

# APLICACIÓN VISUAL MODELUILDER PARA DETERMINAR EL ÍNDICE VULNERABILIDAD A EVENTOS TORRENCIALES

## VISUAL APPLICATION MODEL BUILDER TO DETERMINE THE INDEX VULNERABILITY TO TORRENTIAL EVENTS

Ruth Jimena Vanegas Castellanos

Ingeniera Catastral y Geodesta, Estudiante Especialización en Geomática

Universidad Militar Nueva Granada

Bogotá D.C, Colombia

u3101367@unimilitar.edu.co

### RESUMEN

Esta investigación tuvo por objeto proponer un modelo de aplicación para determinar el índice de vulnerabilidad a eventos torrenciales de una cuenca hidrográfica por medio del cálculo de las características morfométricas de la cuenca y la categorización del grado de vulnerabilidad empleando el ModelBuilder del software ArcGis 10.4, basándose en la metodología para la evaluación regional del agua ERA diseñada por el Instituto de Hidrología, Meteorología y Estudios Ambientales. Para calcular los valores se propone utilizar herramientas de análisis espacial y expresiones condicionales en lenguaje de programación Python, aplicados sobre un Feature Class de área y drenajes de la cuenca, además un Modelo Digital de Elevación y la información correspondiente la variación de los caudales de los últimos años. De la calidad de esta información depende el IVET.

**Palabras clave:** Análisis Espacial, Características Morfométricas, Caudales, Cuenca, DEM, Drenajes, IVET, ModelBuilder, Vulnerabilidad.

### ABSTRACT

This research aimed to propose an application model to determine the vulnerability index to torrential events of a watershed by calculating the morphometric characteristics of the



## UNIVERSIDAD MILITAR NUEVA GRANADA

basin and categorizing the degree of vulnerability, using the ModelBuilder of ArcGis 10.4 software, based on the methodology for the regional evaluation of water ERA designed by the Institute of Hydrology, Meteorology and Environmental Studies. In order to calculate the values it is proposed to use spatial analysis tools and conditional expressions in Python programming language, applied to a Feature Class of area and drainage of the basin, in addition to a Digital Elevation Model and corresponding information the variation of the flows of the last years. The quality of this information depends on the IVET.

**Keywords:** Basin, Drainage, DEM, Flows, IVET, ModelBuilder, Morphometric Characteristics, Spatial Analysis, Vulnerability

### INTRODUCCIÓN

En el Ordenamiento Territorial un componente esencial es la gestión integral del recurso hídrico, dentro de la metodología existente para la evaluación del agua, que analiza aspectos como la oferta, disponibilidad, uso, demanda, contaminación y calidad del agua, así como las amenazas y vulnerabilidad del sistema hídrico, tomando como unidad básica la cuenca hidrográfica. Para el análisis de estos componentes, la metodología propone un sistema de indicadores hídricos que reflejan el estado de cada una de las situaciones del sistema. Cada indicador es una medida cuantitativa o de observación cualitativa con la cual se puede identificar un cambio y el funcionamiento del sistema.

La metodología para la evaluación regional del agua ERA fue diseñada por el Instituto de Hidrología, Meteorología y Estudios Ambientales IDEAM en el año 2013 para indicar lineamientos técnicos a las entidades regionales competentes

para la evaluación regional de los sistemas hídricos. Esta metodología evalúa la oferta, la demanda, la calidad y los riegos a través de indicadores hídricos que reflejan el estado de sistema hídrico (EAAB, 2013). Los indicadores del sistema se dividen en dos, los indicadores del sistema hídrico natural, que son Índice de Retención y Regulación Hídrica (IRH), Índice de Aridez (IA), Índice de Vulnerabilidad Intrínseca a la Contaminación de Agua Subterránea (IVICAS). Y los indicadores de intervención antrópica que son Índice de Uso del Agua superficial (IUA), Índice de Calidad del Agua (ICA), Índice de Calidad Biológica del Agua (IMA), Índice de Alteración potencial de la Calidad de Agua (IACAL), Índice de Vulnerabilidad por desabastecimiento Hídrico (IVH), Índice de Vulnerabilidad a Eventos Torrenciales (IVET) (IDEAM, JULIO 2013).

En virtud de lo anterior, el Índice de Vulnerabilidad a Eventos Torrenciales (IVET) busca categorizar la susceptibilidad de ocurrencia de una



## UNIVERSIDAD MILITAR NUEVA GRANADA

creciente súbita para un cauce de montaña con arrastre de materiales, producida por una fuerte tormenta, denominado evento torrencial de una cuenca hidrográfica, son eventos de corta duración, extremadamente rápidos, pequeña extensión de área de afectación, puede ocurrir en cauces de caudal alto y flujo rápido, y cobertura vegetal escasa. Generalmente causa daños importantes debido a su velocidad y la cantidad de materiales que arrastra. La vulnerabilidad es expresada como la relación existente entre las características físicas de la cuenca (índice morfométrico) y el comportamiento de los caudales (índice de variabilidad) en un periodo de tiempo. La medida del indicador es cualitativa y se expresa por medio de una matriz en términos de vulnerabilidad muy alta, alta, media y baja (IDEAM, JULIO 2013).

El IVET es importante debido a que puede ser un indicador del recurso hídrico en estudios de calidad del agua, evaluación regional del agua, planes de ordenamiento de cuencas hidrográficas pero también puede un indicador en la gestión del riesgo de desastres, su uso puede ayudar a la clasificación del grado de amenaza de una cuenca hidrográfica con lo cual las entidades territoriales y los entes encargados pueden generar programas de prevención y mitigación. Sin embargo dentro de la metodología no está explícito el procedimiento a utilizar para el cálculo, con lo cual se puede generar lentitud en el mismo. Estos problemas se pueden evitar al automatizar el procedimiento para el cálculo del IVET.

El objetivo de este trabajo es proponer un modelo de aplicación para determinar el índice de vulnerabilidad a eventos torrenciales de una cuenca hidrográfica por medio del cálculo de las características morfométricas de la cuenca y la categorización del grado de vulnerabilidad empleando el ModelBuilder del software ArcGis 10.4

La unidad básica de análisis es la cuenca hidrográfica, es un área geográfica limitada por las divisorias de agua, en donde se desarrolla un sistema natural interconectado. Las aguas que vierten en la cuenca pueden ser superficiales o subterráneas, estos flujos integran la escorrentía que alimenta los cauces y las corrientes que generan toda la red hidrográfica. Las cuencas se pueden dividir jerárquicamente en cuencas más pequeñas, su caudal puede ser continuo o intermitente, puede desembocar a un río más grande, a un depósito de agua, o al mar. El límite entre dos cuencas, esto es la divisoria de aguas, es la máxima altura sobre el nivel del mar que bordea la cuenca, dividiendo las aguas lluvias hacia un lado o hacia el otro lado, esta divisoria de aguas se proyecta vertical y horizontalmente, para establecer los modelos hidrográficos de la cuenca.

Según la legislación Colombia el país está dividido en áreas hidrográficas, son las cuencas más grandes del país que agrupan varios ríos y que desembocan al mar; las zonas hidrográficas, son cuencas aquellas cuencas que desembocan sus aguas directamente a un área hidrográfica; subzona hidrográfica, son la cuencas que tributan sus aguas al afluente principal de una zona



## UNIVERSIDAD MILITAR NUEVA GRANADA

hidrográfica; y así sucesivamente para las unidades hidrográficas de nivel I, nivel II y nivel III (IDEAM, NOV 2013). Cada una de las cuales tiene un código y un nombre para facilitar su identificación, el nombre (Toponimia oficial establecida de la base cartográfica del IGAC) depende del río principal, esto es el que tiene el curso más largo y de mayores parámetros morfométricos.

Los parámetros morfométricos de la cuenca tienen que ver con aspectos como el área, la pendiente media, la longitud, el ancho, el coeficiente de compacidad o de forma, entre otros, básicamente características físicas de la cuenca, que son importantes para determinar el comportamiento de la escorrentía, la capacidad de arrastre de los sedimentos así como su velocidad, en si el comportamiento de la cuenca durante un evento catastrófico (Parrado & Romero, 2016).

### 1. MATERIALES Y MÉTODOS

Para la desarrollo de la aplicación se utilizó ModelBuilder del software ArcGis 10.4 bajo la licencia que posee la Universidad Militar. ModelBuilder es un lenguaje de programación visual para crear flujos de trabajo de geoprocésamiento. Estos modelos se automatizan y documentan los procesos de análisis espacial y de administración de datos, son de fácil aplicación debido a que se presentan como un diagrama encadenado de secuencias de procesos y herramientas de análisis espacial, utilizando la salida de un proceso como entrada de otro proceso (ESRI, 2016).

Las herramientas de análisis espacial para geoprocésamiento del software ArcGis 10.4, son utilizadas para el análisis y modelamiento de datos tanto raster como vector, pueden calcular distancias, densidades, interpolación, álgebra de mapas, vecindades, superposiciones, reclasificaciones, modelar flujos, establecer condiciones, generalizar, entre otras.

En el diseño de la aplicación modelbuilder para el cálculo del Índice Vulnerabilidad a Eventos Torrenciales (IVET) se siguió la metodología establecida por el IDEM, tomando los parámetros allí indicados. El IVET está definido como *“la relación existente entre las características de la forma de una cuenca que son indicativos de la torrencialidad en la misma, en relación con las condiciones hidrológicas en dicha cuenca”* (IDEAM, JULIO 2013).

Según lo anterior, las características físicas de la cuenca están indicadas en el índice morfométrico, corresponde a la relación entre los parámetros de la densidad de drenaje, la pendiente media de la cuenca y el coeficiente de compacidad o de forma, como se muestra en la Imagen 1. Para su cálculo se definen valores promedio de las variables indicadas, se establecen seis categorías que indican que a mayor valor hay una mayor tendencia a que en la cuenca se presenten procesos torrenciales. (IDEAM, JULIO 2013).



## UNIVERSIDAD MILITAR NUEVA GRANADA

Índice morfométrico	Escala	Área de la cuenca de drenaje (km <sup>2</sup> )	Categorías				
			1	2	3	4	5
Densidad de drenaje (km/km <sup>2</sup> )	1:10.000	<15	<1,50	1,51 – 2,00	2,01 – 2,50	2,51 – 3,00	> 3
	1:25.000	16 a 50	<1,20	1,21 – 1,80	1,81 – 2,00	2,01 – 2,50	> 2,5
	1:100.000	>50	<1,00	1,01 – 1,50	1,51 – 2,00	2,01 – 2,50	> 2,5
Pendiente media de la cuenca (%)	1:10.000	<15	<20	21 – 35	36 – 50	51 – 75	>75
	1:100.000	>50	<15	16 – 30	30 – 45	46 – 65	>65
			Accidentado	Fuerte	Muy Fuerte	Escarpado	Muy Escarpado
Coeficiente de compacidad			<1,625	1,376 – 1,500	1,251 – 1,375	1,126 – 1,250	1,00 – 1,125
			Oval-oblonga a rectangular-oblonga	Oval-redonda a oval-oblonga	Casi redonda a oval-redonda		

Imagen 1. Relaciones para categorizar el índice morfométrico. Fuente: Instituto de Hidrología, Meteorología y Estudios Ambientales, 2013

Hay que tener en cuenta que, al calcular estos tres parámetros se requiere el uso de otros parámetros como el área, que se define como la proyección horizontal de toda la superficie de escorrentía, corresponde al área delimitada por la divisoria de aguas expresada en kilómetros cuadrados; la pendiente media que representa el relieve característico de la cuenca la superficie de la cuenca, se define como el promedio ponderado de las pendientes internas; la longitud total de los drenajes corresponde a la sumatoria de longitud de los cauces, la longitud de un cauce es la distancia, expresada en kilómetros, desde su nacimiento hasta su desembocadura siguiendo el recorrido de la corriente; el perímetro, que es la longitud sobre un plano horizontal, que recorre la divisoria de aguas, se expresa en kilómetros (Lux, 2016).

Los características físicas de la cuenca, como el área, el perímetro, la altura máxima, la altura mínima, la longitud de los drenajes deben ser determinados

automáticamente con herramientas de análisis espacial del software ArcGis 10.4.

El primero de los parámetros del índice morfométrico de torrencialidad es la densidad de drenaje, corresponde a la relación existente entre la longitud total de los drenajes y el área total indicado en km/km<sup>2</sup>, indicado en la ecuación 1 (Meza, Rodríguez, Corvacho, & Tapia, 2014). Para su cálculo se planteó una expresión en lenguaje de programación visual basic.

$$D_d = \frac{\sum L}{A}$$

Ecuación 1. Densidad de drenaje. Fuente: Meza et al, 2014

El segundo parámetro es la pendiente media de la cuenca, generada automáticamente con el software ArcGis 10.4, tomando como insumo un modelo digital de elevación del terreno con celdas de 12.5 metros.

El tercer parámetro es el coeficiente de compacidad conocido como índice de Gravelius, compara el área de la cuenca con un círculo de igual área que la cuenca, cuando los valores del coeficiente son cercanos a uno se dice que la cuenca es circular con cauces torrenciales, fuertes volúmenes de aguas de escurrimiento y tiempos de concentración cortos; cuando los valores están entre 1 y 1,25 la cuenca se considera redonda a oval redonda; si esta entre 1,25 y 1,5 la cuenca es de oval redonda a oval oblonga y por ultimo si el valor de coeficiente esta entre 1,5 y 1,75 la cuenca se considera de oval oblonga a





## UNIVERSIDAD MILITAR NUEVA GRANADA

rectangular oblonga. El valor del coeficiente de compacidad es adimensional y nunca será menor que uno (Huertas, 2016). Se calcula a través de la siguiente ecuación 2, en el modelo se calcula utilizando una expresión en lenguaje de programación Python.

$$I_g = \frac{\text{Perimetro de la cuenca}}{2 (\pi \text{ Area})^{0,5}}$$

*Ecuación 2. Coeficiente de compacidad. Fuente: Huertas, 2016*

Una vez calculados los valores de cada uno de los parámetros categorizan con base en imagen 1 relaciones para categorizar el índice morfométrico, estos valores fueron establecidos mediante expresiones condicionales en lenguaje de programación Python en el modelo.

Seguidamente, mediante expresiones condicionales, realiza la categorización de índice morfométrico de torrencialidad en función de los parámetros anteriores, el cual se establece según las relaciones establecidas en la imagen 2. (IDEAM, JULIO 2013).

		Pendiente media de la cuenca					
		1	2	3	4	5	
Densidad de drenaje	1	111	121	131	141	151	1
		112	122	132	142	152	2
		113	123	133	143	153	3
		114	124	134	144	154	4
		115	125	135	145	155	5
	2	211	221	231	241	251	1
		212	222	232	242	252	2
		213	223	233	243	253	3
		214	224	234	244	254	4
		215	225	235	245	255	5
	3	311	321	331	341	351	1
		312	322	332	342	352	2
		313	323	333	343	353	3
		314	324	334	344	354	4
		315	325	335	345	355	5
	4	411	421	431	441	451	1
		412	422	432	442	452	2
		413	423	433	443	453	3
		414	424	434	444	454	4
		415	425	435	445	455	5
	5	511	521	531	541	551	1
		512	522	532	542	552	2
		513	523	533	543	553	3
		514	524	534	544	554	4
		515	525	535	545	555	5

■ Muy alta    ■ Baja    ■ Muy baja  
■ Alta    ■ Moderada

*Imagen 2. Relaciones entre variables para el índice morfométrico. Fuente: Instituto de Hidrología, Meteorología y Estudios Ambientales, 2013*

El índice de variabilidad indica la variabilidad de los caudales en una determinada cuenca. Se calcula generando una curva de duración de caudales, es una relación logarítmica de dos lecturas de caudales tomados en dos épocas distintas. Una cuenca donde existan grandes diferencias entre los valores mínimos y máximos, tiene mucha variabilidad se puede afirmar que es cuenca torrencial (IDEAM, JULIO 2013).

La categorización del índice de variabilidad se realiza comparando con expresiones condicionales los porcentajes de variabilidad de la curva de duración de caudales a escala logarítmica.



# UNIVERSIDAD MILITAR NUEVA GRANADA

Índice de variabilidad	Vulnerabilidad
< 10°	Muy baja
10.1° - 37°	Baja
37.1° - 47°	Media
47.1° - 55°	Alta
>55°	Muy alta

Imagen 3. Clasificación del índice de variabilidad.  
Fuente: Instituto de Hidrología, Meteorología y Estudios Ambientales, 2013

Finalmente se estima el índice de vulnerabilidad a eventos torrenciales teniendo en cuenta la categorización del índice morfométrico de torrencialidad y el índice de variabilidad. También se calcula por medio de expresiones condicionales en lenguaje de programación Python, la clasificaciones están indicadas en la imagen 4.

Índice de variabilidad	Índice morfométrico de torrencialidad				
	Muy baja	Baja	Media	Alta	Muy alta
Muy baja	Baja	Baja	Media	Alta	Alta
Baja	Baja	Media	Media	Alta	Muy alta
Media	Baja	Media	Alta	Alta	Muy alta
Alta	Media	Media	Alta	Muy alta	Muy alta
Muy alta	Media	Alta	Alta	Muy alta	Muy alta

Imagen 4. Clasificación del índice de vulnerabilidad frente a eventos torrenciales.  
Fuente: Instituto de Hidrología, Meteorología y Estudios Ambientales, 2013

## 1.1 MATERIALES

Para la ejecución del modelo es necesario un Modelo Digital de Elevación, un Feature Class tipo línea de los drenajes, un Feature Class tipo polígono con la cuenca, donde se ingresaran los valores generados; y una base de datos con información de caudales diarios o estimados del área de la cuenca.

## 2. RESULTADOS

Para facilitar la comprensión del modelo se presentaran los resultados en pequeñas partes. En el cálculo de las características físicas de la cuenca como el área, la altura máxima, la altura mínima se utilizaron estadísticas en una tabla, que posteriormente se unirá al Feature Class. El perímetro fue determinado con una herramienta de análisis espacial para calcular geometrías.

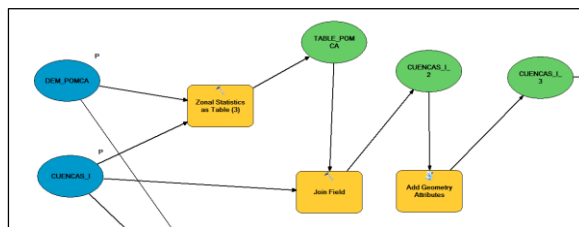


Imagen 5. ModelBuilder para el cálculo de características físicas

La longitud total de los drenajes se calcula tomado como insumo un feature tipo línea a escala 2:25000. Posteriormente se ingresara al feature class como un valor constante.

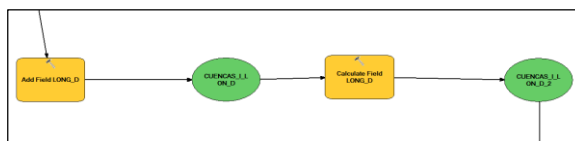


Imagen 6. ModelBuilder para longitud total de drenajes

El factor de forma fue calculado dividiendo el área en la longitud de los drenajes al cuadrado. Este compara el límite de la cuenca con un ovoide, su valor indica que tan alargada o redonda es la





## UNIVERSIDAD MILITAR NUEVA GRANADA

cuenca, valores mayores a uno para cuencas alargadas y menores a uno para cuencas redondas.

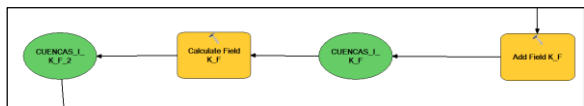


Imagen 7. ModelBuilder para el factor de forma

La densidad de drenaje fue calculada con la formula indicada en la ecuación 1, posteriormente se categorizo utilizando un expresión condicional en lenguaje de programación Python, como lo muestra la imagen 9.

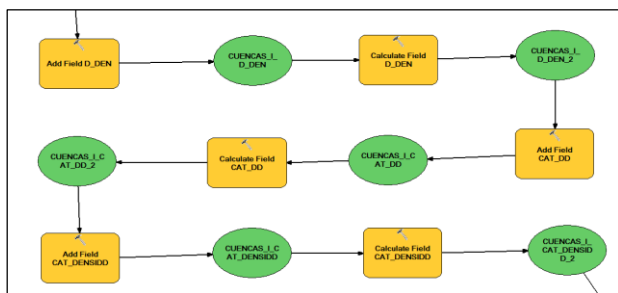


Imagen 8. ModelBuilder para densidad de drenaje

```
def comparacion(cat_dd):
    if cat_dd < 1.20:
        return 1
    elif cat_dd >= 1.21 and cat_dd<=1.80:
        return 2
    elif cat_dd >= 1.81 and cat_dd<=2.00:
        return 3
    elif cat_dd >= 2.01 and cat_dd<=2.50:
        return 4
    elif cat_dd >2.5:
        return 5
    else:
        return 0
```

Imagen 9. Categorización Densidad de Drenaje

Según la metodología, la categorización densidad de drenaje tiene descripción, se

utilizó una expresión condicional en lenguaje de programación Python, como se muestra en la imagen 10, para su definición.

```
def Comparacion(cat_dd):
    if cat_dd==1:
        return "baja"
    elif cat_dd==2:
        return "moderada"
    elif cat_dd==3:
        return "moderada alta"
    elif cat_dd==4:
        return "alta"
    elif cat_dd==5:
        return "muy alta"
    else:
        return "error"
```

Imagen 10. Definición de categorización de la Densidad de Drenaje

Para determinar de la pendiente media, imagen 11, se calculó con herramientas del software la pendiente, seguidamente se realiza un proceso de análisis espaciales y estadísticas para calcular la media. Se categoriza y se realiza su descripción utilizando el mismo criterio que el utilizado con densidad de drenaje

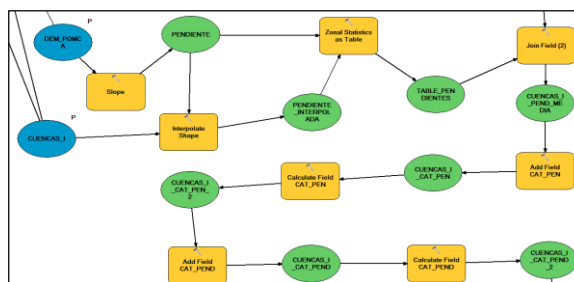
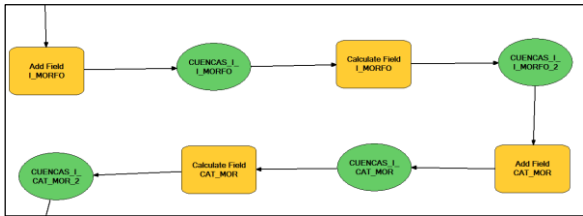


Imagen 11. ModelBuilder para pendiente media de la cuenca

El coeficiente de compacidad, imagen 12, se calcula utilizando la ecuación 2 y se categoriza y se describe utilizando el

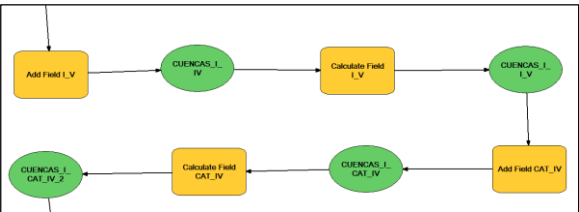
[illegible]

Una vez calculados, categorizados y descritos los tres parámetros de índice morfométrico se procede a calcular su clasificación imagen 13, primero se unifican las categorías de los parámetros en un número, posteriormente se compara mediante expresión condicional en lenguaje de programación Python, indicada en la imagen 14, con la tabla de relaciones entre variables para el índice morfométrico imagen 2.



```
def Comparacion(avg):
    if avg==111 and avg<=114:
        return "Muy Baja"
    elif avg==121 or avg==122:
        return "Muy Baja"
    elif avg==115:
        return "Baja"
    elif avg==123 and avg<=125:
        return "Baja"
    elif avg==131 and avg<=135:
        return "Baja"
    elif avg==141 and avg<=145:
        return "Baja"
    elif avg==151 and avg<=155:
        return "Baja"
    elif avg==211 and avg<=215:
        return "Baja"
    elif avg==221 and avg<=225:
        return "Baja"
    elif avg==231 and avg<=231:
        return "Baja"
    elif avg==241:
        return "Baja"
    elif avg==311 and avg<=314:
        return "baja"
    elif avg==321 and avg<=322:
        return "Baja"
    elif avg==315:
        return "Moderada"
    elif avg==323 and avg<=325:
        return "Moderada"
    elif avg==331 and avg<=335:
        return "Moderada"
    elif avg==334 and avg<=335:
        return "Moderada"
    elif avg==242 and avg<=245:
        return "Moderada"
    elif avg==251 and avg<=255:
        return "Moderada"
    elif avg==242 and avg<=245:
        return "Moderada"
    elif avg==341 and avg<=344:
        return "Moderada"
    elif avg==351 and avg<=353:
        return "Moderada"
    elif avg==411 and avg<=415:
        return "Moderada"
    elif avg==421 and avg<=424:
        return "Moderada"
    elif avg==431 and avg<=433:
        return "Moderada"
    elif avg==511 and avg<=514:
        return "Moderada"
    elif avg==515:
        return "Alta"
    elif avg==425:
        return "Alta"
    elif avg==521 and avg<=525:
        return "Alta"
    elif avg==434 and avg<=435:
        return "Alta"
    elif avg==345:
        return "Alta"
    elif avg==441 and avg<=445:
        return "Alta"
    elif avg==354 and avg<=355:
        return "Alta"
    elif avg==451 and avg<=454:
        return "Alta"
    elif avg==531 and avg<=535:
        return "Alta"
    elif avg==541 and avg<=543:
        return "Alta"
    elif avg==544 and avg<=545:
        return "Muy Alta"
    elif avg==455:
        return "Muy Alta"
    elif avg==551 and avg<=555:
        return "Muy Alta"
    else:
        return "error"
```

Para determinar el índice de variabilidad se requiere dibujar las curvas de variabilidad a escala logarítmica y calcular la pendiente de la misma. Se decidió, calcular las curvas de variabilidad con otro software e ingresar los valores como constantes. La categorización del índice de variabilidad se realizó utilizando expresión condicional en lenguaje de programación Python imagen 15.



La clasificación del IVET, imagen 16, se elaboró una expresión condicional en lenguaje de programación Python imagen 17, tomado como referencia la metodología del IDEAM.



## UNIVERSIDAD MILITAR NUEVA GRANADA

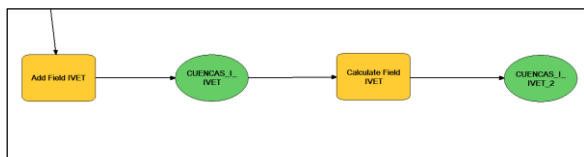


Imagen 16. ModelBuilder para el Índice de vulnerabilidad frente evento torrenciales (IVET)

```
def Comparacion(iv,im):
if iv=="Muy Baja" and im=="Muy baja":
return "Baja"
elif iv=="Muy Baja" and im=="Baja":
return "Baja"
elif iv=="Muy Baja" and im=="Moderada":
return "Media"
elif iv=="Muy Baja" and im=="Alta":
return "Alta"
elif iv=="Muy Baja" and im=="Muy Alta":
return "Baja"
elif iv=="Baja" and im=="Muy Baja":
return "Baja"
elif iv=="Baja" and im=="Baja":
return "Media"
elif iv=="Baja" and im=="Moderada":
return "Media"
elif iv=="Baja" and im=="Alta":
return "Alta"
elif iv=="Baja" and im=="Muy Alta":
return "Muy Alta"
elif iv=="Media" and im=="Muy Baja":
return "Baja"
elif iv=="Media" and im=="Baja":
return "Media"
elif iv=="Media" and im=="Moderada":
return "Alta"
elif iv=="Media" and im=="Alta":
return "Muy Alta"
elif iv=="Media" and im=="Muy Alta":
return "Muy Alta"
elif iv=="Alta" and im=="Muy Baja":
return "Baja"
elif iv=="Alta" and im=="Baja":
return "Media"
elif iv=="Alta" and im=="Moderada":
return "Alta"
elif iv=="Alta" and im=="Alta":
return "Muy Alta"
elif iv=="Alta" and im=="Muy Alta":
return "Muy Alta"
elif iv=="Muy Alta" and im=="Muy Baja":
return "Baja"
elif iv=="Muy Alta" and im=="Baja":
return "Alta"
elif iv=="Muy Alta" and im=="Moderada":
return "Alta"
elif iv=="Muy Alta" and im=="Alta":
return "Muy Alta"
elif iv=="Muy Alta" and im=="Muy Alta":
return "Muy Alta"
else:
return "error"
```

Imagen 17. Expresión condicional para el Índice Morfométrico

La imagen 18 representa estructura general del modelo de aplicación para determinar el índice de vulnerabilidad a eventos torrenciales de una cuenca hidrográfica.

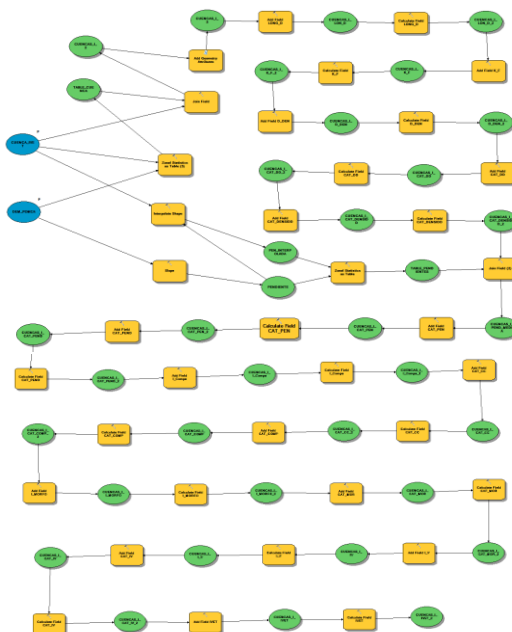


Imagen 18. Representación general del modelo

### 3. CONCLUSIONES Y RECOMENDACIONES

- El ModelBuilder del software ArcGis 10.4 propuesto, permite determinar el índice de vulnerabilidad a eventos torrenciales de una cuenca hidrográfica, sin embargo es necesario tener presente que algunos cálculos no fue posible automatizarlos, debido principalmente a la dificultad de integrarlos al modelo, como la longitud total de los drenajes y índice de variabilidad de los drenajes. Además se debe tener en cuenta la unidad de medida de cada uno de los parámetros.



## UNIVERSIDAD MILITAR NUEVA GRANADA

- Los parámetros y los insumos para el diseño de la aplicación ModelBuilder del software ArcGis 10.4 fueron seleccionados de acuerdo a la metodología de IDEAM, pero fue necesario adaptarlos debido a que la metodología presenta algunos errores, de concordancia en la categorización del índice morfométrico y del índice de variabilidad.
- La formulación de la aplicación en lenguaje de programación visual de un modelo para el cálculo del IVET se elaboró con herramientas de análisis espacial de fácil manejo y aplicación, como expresiones condicionales, cálculos de geometrías como áreas, perímetros.
- La verificación de modelo se realizó con la cuenca del río la Ceibas y la cuenca del río Loro ubicadas en el departamento del Huila, un DEM Alos Palsar con resolución de 12.5 mts y la información de variación de caudales. La clasificación del IVET arrojó un índice Muy Alto, según el Consejo Departamental para la Gestión del Riesgo de Desastres del Huila, históricamente estas dos cuencas tienen antecedentes de avenidas torrenciales frecuentes.
- La calidad de la categorización depende de la calidad de los insumos, un DEM alta resolución, un Feature Class de drenajes y del área de la cuenca a escala grande.
- En el software ArcGis existe una herramienta denominada Hidrology, con ella se puede generar un Feature Class de drenajes y de cuencas y subcuencas, sin embargo es necesario que estos sean revisados por un experto en hidrología.
- En la Gestión del Riesgo se evalúan amenazas y vulnerabilidades, uno de los elementos a valorar es el recurso hídrico, debido a la importancia que tiene para una comunidad, por tanto, automatizar procesos para determinar vulnerabilidades como el IVET facilita su clasificación y minimiza posibles errores.

## BIBLIOGRAFÍA

- EAAB. (2013). *Metodología para la Evaluación Regional del Agua (ERA)*. Bogotá D.C: Acueducto, Agua, Alcantarillado y Aseo de Bogotá.
- ESRI. (2016). *ARCGIS PRO*. Obtenido de <https://pro.arcgis.com/es/pro-app/help/analysis/geoprocessing.htm>
- Huertas, U. E. (2016). *Evaluación de la Vulnerabilidad y amenaza de los sistemas hídricos en el marco de la evaluación regional del agua para la cuenca del río Sumapa, jurisdicción CAR*. Bogotá D.C: Universidad Distrital Francisco José de Caldas.
- IDEAM. (JULIO 2013). *Lineamientos Conceptuales y Metodológicos para la Evaluación del Agua - ERA*. Bogotá D.C.
- IDEAM. (NOV 2013). *Zonificación y codificación de unidades hidrográficas e*



**UNIVERSIDAD MILITAR  
NUEVA GRANADA**

*hidrogeológicas de Colombia*. Bogotá  
D.C.: IDEAM.

LEY 1523. (2012). LEY 1523 DE 2012 "Por la cual se adopta la política nacional de gestión del riesgo de desastres y se establece el Sistema Nacional de Gestión del Riesgo de Desastres y se dictan otras disposiciones". Bogotá: CONGRESO DE LA REPUBLICA.

Lux, C. B. (2016). *Conceptos básicos de Morfometría de Cuencas Hidrográficas*. Universidad de San Carlos de Guatemala.

Meza, A. M., Rodríguez, V. A., Corvacho, G. O., & Tapia, T. A. (2014). Análisis morfométrico de microcuencas afectadas por flujos de detritos bajo precipitación intensa en la Quebrada de Camuña, Norte Grande de Chile. *Diálogo Andino*.

Parrado, T. D., & Romero, D. A. (2016). *Evaluación del riesgo al recurso hídrico en la cuenca Ubaté- Suárez, enmarcado en los lineamientos conceptuales y metodológicos para la evaluación regional del agua - ERA*. Bogotá D.C.: Universidad Distrital Francisco José de Caldas.



## UNIVERSIDAD MILITAR NUEVA GRANADA

### ANEXO 1. Scrit Modelbuilder en lenguaje de programación Python

```
# Import arcpy module
import arcpy

# Check out any necessary licenses
arcpy.CheckOutExtension("spatial")
arcpy.CheckOutExtension("3D")
# Script arguments
DEM_POMCA = arcpy.GetParameterAsText(0)
if DEM_POMCA == '#' or not DEM_POMCA:
    DEM_POMCA = "G:\\PROYECTO\\info_espacial\\CUENCAS.gdb\\DEM_POMCA" # provide a default value if
    unspecified

CUENCA_IVET = arcpy.GetParameterAsText(1)
if CUENCA_IVET == '#' or not CUENCA_IVET:
    CUENCA_IVET = "CUENCA_IVET" # provide a default value if unspecified

# Local variables:
TABLE_CUENCA = DEM_POMCA
CUENCAS_I_2 = TABLE_CUENCA
CUENCAS_I_3 = CUENCAS_I_2
CUENCAS_I_LON_D = CUENCAS_I_3
CUENCAS_I_LON_D_2 = CUENCAS_I_LON_D
CUENCAS_I_K_F = CUENCAS_I_LON_D_2
CUENCAS_I_K_F_2 = CUENCAS_I_K_F
CUENCAS_I_D_DEN = CUENCAS_I_K_F_2
CUENCAS_I_D_DEN_2 = CUENCAS_I_D_DEN
CUENCAS_I_CAT_DD = CUENCAS_I_D_DEN_2
CUENCAS_I_CAT_DD_2 = CUENCAS_I_CAT_DD
CUENCAS_I_CAT_DENSIDD = CUENCAS_I_CAT_DD_2
CUENCAS_I_CAT_DENSIDD_2 = CUENCAS_I_CAT_DENSIDD
CUENCAS_I_PEND_MEDIA = CUENCAS_I_CAT_DENSIDD_2
CUENCAS_I_CAT_PEN = CUENCAS_I_PEND_MEDIA
CUENCAS_I_CAT_PEN_2 = CUENCAS_I_CAT_PEN
CUENCAS_I_CAT_PEND = CUENCAS_I_CAT_PEN_2
CUENCAS_I_CAT_PEND_2 = CUENCAS_I_CAT_PEND
CUENCAS_I_I_Comp = CUENCAS_I_CAT_PEND_2
CUENCAS_I_I_Comp_2 = CUENCAS_I_I_Comp
CUENCAS_I_CAT_CC = CUENCAS_I_I_Comp_2
CUENCAS_I_CAT_CC_2 = CUENCAS_I_CAT_CC
CUENCAS_I_CAT_COMP = CUENCAS_I_CAT_CC_2
CUENCAS_I_CAT_COMP_2 = CUENCAS_I_CAT_COMP
CUENCAS_I_I_MORFO = CUENCAS_I_CAT_COMP_2
CUENCAS_I_I_MORFO_2 = CUENCAS_I_I_MORFO
CUENCAS_I_CAT_MOR = CUENCAS_I_I_MORFO_2
CUENCAS_I_CAT_MOR_2 = CUENCAS_I_CAT_MOR
CUENCAS_I_IV = CUENCAS_I_CAT_MOR_2
CUENCAS_I_I_V = CUENCAS_I_IV
CUENCAS_I_CAT_IV = CUENCAS_I_I_V
CUENCAS_I_CAT_IV_2 = CUENCAS_I_CAT_IV
CUENCAS_I_IVET = CUENCAS_I_CAT_IV_2
CUENCAS_I_IVET_2 = CUENCAS_I_IVET
PENDIENTE = DEM_POMCA
PEN_INTERPOLADA = PENDIENTE
TABLE_PENDIENTES = PEN_INTERPOLADA

# Process: Zonal Statistics as Table (3)
```





## UNIVERSIDAD MILITAR NUEVA GRANADA

```
arcpy.gp.ZonalStatisticsAsTable_sa(CUENCA_IVET, "Hoyas", DEM_POMCA, TABLE_CUENCA, "DATA", "ALL")
```

```
# Process: Join Field
```

```
arcpy.JoinField_management(CUENCA_IVET, "Hoyas", TABLE_CUENCA, "Hoyas", "Area;Min;Max;Std")
```

```
# Process: Add Geometry Attributes
```

```
arcpy.AddGeometryAttributes_management(CUENCAS_I_2, "PERIMETER_LENGTH", "KILOMETERS",  
"SQUARE_KILOMETERS", "")
```

```
# Process: Add Field LONG_D
```

```
arcpy.AddField_management(CUENCAS_I_3, "LONG_D", "DOUBLE", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

```
# Process: Calculate Field LONG_D
```

```
arcpy.CalculateField_management(CUENCAS_I_LON_D, "LONG_D", "Comparacion( !Hoyas! )", "PYTHON_9.3",  
"def Comparacion(avg): \n if avg == \"RIO_CEIBAS\":\n return 1095.377\n elif avg == \"RIO_LORO\":\n  
return 1192.329\n else:\n return 0")
```

```
# Process: Add Field K_F
```

```
arcpy.AddField_management(CUENCAS_I_LON_D_2, "K_F", "DOUBLE", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

```
# Process: Calculate Field K_F
```

```
arcpy.CalculateField_management(CUENCAS_I_K_F, "K_F", "[AREA] / ([LONG_D] * [LONG_D])", "VB", "")
```

```
# Process: Add Field D_DEN
```

```
arcpy.AddField_management(CUENCAS_I_K_F_2, "D_DEN", "DOUBLE", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

```
# Process: Calculate Field D_DEN
```

```
arcpy.CalculateField_management(CUENCAS_I_D_DEN, "D_DEN", "[LONG_D] / [AREA]", "VB", "")
```

```
# Process: Add Field CAT_DD
```

```
arcpy.AddField_management(CUENCAS_I_D_DEN_2, "CAT_DD", "SHORT", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

```
# Process: Calculate Field CAT_DD
```

```
arcpy.CalculateField_management(CUENCAS_I_CAT_DD, "CAT_DD", "Comparacion( !D_DEN! )", "PYTHON", "def  
Comparacion(cat_dd): \n if cat_dd < 1.20:\n return 1\n elif cat_dd >= 1.21 and cat_dd <= 1.80:\n return 2\n  
elif cat_dd >= 1.81 and cat_dd <= 2.00:\n return 3\n elif cat_dd >= 2.01 and cat_dd <= 2.50:\n return 4\n elif  
cat_dd > 2.5:\n return 5\n else:\n return 0")
```

```
# Process: Add Field CAT_DENSIDD
```

```
arcpy.AddField_management(CUENCAS_I_CAT_DD_2, "CAT_DENSIDD", "TEXT", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

```
# Process: Calculate Field CAT_DENSIDD
```

```
arcpy.CalculateField_management(CUENCAS_I_CAT_DENSIDD, "CAT_DENSIDD", "Comparacion( !CAT_DD! )",  
"PYTHON", "def Comparacion(avg): \n if avg==1:\n return \"baja\"\n elif avg==2:\n return \"moderada\"\n elif  
avg==3:\n return \"moderada alta\"\n elif avg==4:\n return \"alta\"\n elif avg==5:\n return \"muy alta\"\n  
else:\n return \"error\"")
```

```
# Process: Slope
```

```
arcpy.gp.Slope_sa(DEM_POMCA, PENDIENTE, "PERCENT_RISE", "1")
```

```
# Process: Interpolate Shape
```

```
arcpy.InterpolateShape_3d(PENDIENTE, CUENCA_IVET, PEN_INTERPOLADA, "", "1", "BILINEAR", "DENSIFY",  
"0")
```

```
# Process: Zonal Statistics as Table
```



## UNIVERSIDAD MILITAR NUEVA GRANADA

```
arcpy.gp.ZonalStatisticsAsTable_sa(PEN_INTERPOLADA, "Hoyas", PENDIENTE, TABLE_PENDIENTES, "DATA",  
"ALL")
```

# Process: Join Field (2)

```
arcpy.JoinField_management(CUENCAS_I_CAT_DENSIDD_2, "Hoyas", TABLE_PENDIENTES, "Hoyas", "MEAN")
```

# Process: Add Field CAT\_PEN

```
arcpy.AddField_management(CUENCAS_I_PEND_MEDIA, "CAT_PEN", "SHORT", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

# Process: Calculate Field CAT\_PEN

```
arcpy.CalculateField_management(CUENCAS_I_CAT_PEN, "CAT_PEN", "Comparacion( !MEAN! )", "PYTHON",  
"def Comparacion(avg): \n if avg <=15:\n return 1\n elif avg > 15 and avg<=30:\n return 2\n elif avg > 30  
and avg<=45:\n return 3\n elif avg > 45 and avg<=65:\n return 4\n elif avg > 65:\n return 5\n else:\n  
return 0")
```

# Process: Add Field CAT\_PEND

```
arcpy.AddField_management(CUENCAS_I_CAT_PEN_2, "CAT_PEND", "TEXT", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

# Process: Calculate Field CAT\_PEND

```
arcpy.CalculateField_management(CUENCAS_I_CAT_PEND, "CAT_PEND", "Comparacion( !CAT_PEN! )",  
"PYTHON", "def Comparacion(avg): \n if avg==1:\n return \"accidentado\"\n elif avg==2:\n return \"fuerte\"\n  
elif avg==3:\n return \"muy fuerte\"\n elif avg==4:\n return \"escarpado\"\n elif avg==5:\n return \"muy  
escarpado\"\n else:\n return \"error\"")
```

# Process: Add Field I\_Comp

```
arcpy.AddField_management(CUENCAS_I_CAT_PEND_2, "I_comp", "DOUBLE", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

# Process: Calculate Field I\_Comp

```
arcpy.CalculateField_management(CUENCAS_I_I_Comp, "I_comp", "!PERIMETER! /(2 *(( math.pi * !AREA! )**  
0.5))", "PYTHON", "")
```

# Process: Add Field CAT\_CC

```
arcpy.AddField_management(CUENCAS_I_I_Comp_2, "CAT_CC", "SHORT", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

# Process: Calculate Field CAT\_CC

```
arcpy.CalculateField_management(CUENCAS_I_CAT_CC, "CAT_CC", "Comparacion( !I_comp! )", "PYTHON", "def  
Comparacion(avg): \n if avg >= 1 and avg<=1.125:\n return 5\n elif avg >= 1.126 and avg<=1.250:\n return  
4\n elif avg >= 1.251 and avg<=1.375:\n return 3\n elif avg >= 1.376 and avg<=1.500:\n return 2\n elif avg  
> 1.500:\n return 1\n else:\n return 0")
```

# Process: Add Field CAT\_COMP

```
arcpy.AddField_management(CUENCAS_I_CAT_CC_2, "CAT_COMP", "TEXT", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

# Process: Calculate Field CAT\_COMP

```
arcpy.CalculateField_management(CUENCAS_I_CAT_COMP, "CAT_COMP", "Comparacion( !CAT_CC! )",  
"PYTHON", "def Comparacion(avg): \n if avg==1:\n return \"oval-oblonga a rectangular-oblonga\"\n elif avg==2  
or avg==3:\n return \"oval-redonda a oval-oblonga\"\n elif avg==4 or avg==5:\n return \"casi redonda a oval-  
redonda\"\n else:\n return \"error\"")
```

# Process: Add Field I\_MORFO

```
arcpy.AddField_management(CUENCAS_I_CAT_COMP_2, "I_MORFO", "LONG", "", "", "", "", "NULLABLE",  
"NON_REQUIRED", "")
```

# Process: Calculate Field I\_MORFO



## UNIVERSIDAD MILITAR NUEVA GRANADA

```
arcpy.CalculateField_management(CUENCAS_I_I_MORFO, "I_MORFO", "[CAT_DD] & \"\" & [CAT_PEN] & \"\" & [CAT_CC]", "VB", "")
```

# Process: Add Field CAT\_MOR

```
arcpy.AddField_management(CUENCAS_I_I_MORFO_2, "CAT_MOR", "TEXT", "", "", "", "", "NULLABLE", "NON_REQUIRED", "")
```

# Process: Calculate Field CAT\_MOR

```
arcpy.CalculateField_management(CUENCAS_I_CAT_MOR, "CAT_MOR", "Comparacion( !I_MORFO! )", "PYTHON", "def Comparacion(avg): \n if avg==111 and avg<=114:\n return \"Muy Baja\"\n elif avg==121 or avg==122:\n return \"Muy Baja\"\n elif avg==115:\n return \"Baja\"\n elif avg==123 and avg<=125:\n return \"Baja\"\n elif avg==131 and avg<=135:\n return \"Baja\"\n elif avg==141 and avg<=145:\n return \"Baja\"\n elif avg==151 and avg<=155:\n return \"Baja\"\n elif avg==211 and avg<=215:\n return \"Baja\"\n elif avg==221 and avg<=225:\n return \"Baja\"\n elif avg==231 and avg<=231:\n return \"Baja\"\n elif avg==241:\n return \"Baja\"\n elif avg==311 and avg<=314:\n return \"Baja\"\n elif avg==321 and avg<=322:\n return \"Baja\"\n elif avg==315:\n return \"Moderada\"\n elif avg==323 and avg<=325:\n return \"Moderada\"\n elif avg==331 and avg<=335:\n return \"Moderada\"\n elif avg==234 and avg<=235:\n return \"Moderada\"\n elif avg==242 and avg<=245:\n return \"Moderada\"\n elif avg==251 and avg<=255:\n return \"Moderada\"\n elif avg==242 and avg<=245:\n return \"Moderada\"\n elif avg==341 and avg<=344:\n return \"Moderada\"\n elif avg==351 and avg<=353:\n return \"Moderada\"\n elif avg==411 and avg<=415:\n return \"Moderada\"\n elif avg==421 and avg<=424:\n return \"Moderada\"\n elif avg==431 and avg<=433:\n return \"Moderada\"\n elif avg==511 and avg<=514:\n return \"Moderada\"\n elif avg==515:\n return \"Alta\"\n elif avg==425:\n return \"Alta\"\n elif avg==521 and avg<=525:\n return \"Alta\"\n elif avg==434 and avg<=435:\n return \"Alta\"\n elif avg==345:\n return \"Alta\"\n elif avg==441 and avg<=445:\n return \"Alta\"\n elif avg==354 and avg<=355:\n return \"Alta\"\n elif avg==451 and avg<=454:\n return \"Alta\"\n elif avg==531 and avg<=535:\n return \"Alta\"\n elif avg==541 and avg<=543:\n return \"Alta\"\n elif avg==544 and avg<=545:\n return \"Muy Alta\"\n elif avg==455:\n return \"Muy Alta\"\n elif avg==551 and avg<=555:\n return \"Muy Alta\"\n else:\n return \"error\"")
```

# Process: Add Field I\_V

```
arcpy.AddField_management(CUENCAS_I_CAT_MOR_2, "I_V", "DOUBLE", "", "", "", "", "NULLABLE", "NON_REQUIRED", "")
```

# Process: Calculate Field I\_V

```
arcpy.CalculateField_management(CUENCAS_I_IV, "I_V", "Comparacion( !Hoyas! )", "PYTHON", "def Comparacion(avg): \n if avg == \"RIO_CEIBAS\":\n return 56\n elif avg == \"RIO_LORO\":\n return 88\n else:\n return 0")
```

# Process: Add Field CAT\_IV

```
arcpy.AddField_management(CUENCAS_I_I_V, "CAT_IV", "TEXT", "", "", "", "", "NULLABLE", "NON_REQUIRED", "")
```

# Process: Calculate Field CAT\_IV

```
arcpy.CalculateField_management(CUENCAS_I_CAT_IV, "CAT_IV", "Comparacion( !I_V! )", "PYTHON", "def Comparacion(iv): \n if iv<=10:\n return \"Muy Baja\"\n elif iv >=10.1 and iv <=37:\n return \"Baja\"\n elif iv>=37.1 and iv <=47:\n return \"media\"\n elif iv >=47.1 and iv >=55:\n return \"Alta\"\n elif iv>55:\n return \"Muy Alta\"\n else:\n return \"error\"")
```

# Process: Add Field IVET

```
arcpy.AddField_management(CUENCAS_I_CAT_IV_2, "IVET", "TEXT", "", "", "", "", "NULLABLE", "NON_REQUIRED", "")
```

# Process: Calculate Field IVET

```
arcpy.CalculateField_management(CUENCAS_I_IVET, "IVET", "Comparacion( !CAT_IV! , !CAT_MOR! )", "PYTHON", "def Comparacion(iv,im): \n if iv==\"Muy Baja\" and im==\"Muy baja\":\n return \"Baja\"\n elif iv==\"Muy Baja\" and im==\"Baja\":\n return \"Baja\"\n elif iv==\"Muy Baja\" and im==\"Moderada\":\n return \"Media\"\n elif iv==\"Muy Baja\" and im==\"Alta\":\n return \"Alta\"\n elif iv==\"Muy Baja\" and im==\"Muy Alta\":\n return \"Alta\"\n elif iv==\"Baja\" and im==\"Muy Baja\":\n return \"Baja\"\n elif iv==\"Baja\" and im==\"Baja\":\n return \"Media\"\n elif iv==\"Baja\" and im==\"Moderada\":\n return \"Media\"\n elif iv==\"Baja\" and im==\"Alta\":\n return \"Alta\"\n elif iv==\"Baja\" and im==\"Muy Alta\":\n return \"Muy Alta\"\n elif
```



## UNIVERSIDAD MILITAR NUEVA GRANADA

```
iv=="Medial" and im=="Muy Baja":\n    return "Baja"\n elif iv=="Medial" and im=="Baja":\n    return "Medial"\n elif iv=="Medial" and im=="Moderada":\n    return "Alta"\n elif iv=="Medial" and im=="Alta":\n    return "Alta"\n elif iv=="Medial" and im=="Muy Alta":\n    return "Muy Alta"\n elif iv=="Alta" and im=="Muy Baja":\n    return "Medial"\n elif iv=="Alta" and im=="Moderada":\n    return "Alta"\n elif iv=="Alta" and im=="Alta":\n    return "Muy Alta"\n elif iv=="Alta" and im=="Muy Alta":\n    return "Muy Alta"\n elif iv=="Muy Alta" and im=="Muy Baja":\n    return "Medial"\n elif iv=="Muy Alta" and im=="Baja":\n    return "Alta"\n elif iv=="Muy Alta" and im=="Moderada":\n    return "Alta"\n elif iv=="Muy Alta" and im=="Alta":\n    return "Muy Alta"\n elif iv=="Muy Alta" and im=="Muy Alta":\n    return "Muy Alta"\n else:\n    return "error")
```